SOFTWARE

Process modeling software proves to be a solution to a different software problem: packaging the disks properly.

SEEING ELECTRONICALLY IS BELIEVING

BY LAWRENCE GOULD

Printing giant R.R. Donnelley & Sons Co. (Crawfordsville, IN) purchased a custom-built machine essential to its production: a high-speed software-disk collator for retail packaging. It wasn't prepared, however, for three problems that arose from that purchase.

Problem 1: The collator's manufacturer closed down while the machine was being installed.

Problem 2: The collator operated at less than half its designed capacity, due to timing difficulties. Various machine builders recommended remedies to increase the capacity of the collator. The costs ranged from $20,000 to $500,000.

Problem 3: Which remedy should Donnelley choose?

To solve the selection problem, the company designed a collator model—in software. Then the effect of each remedy was simulated in real time. This process eliminated the trial-and-error process of most evaluations.

Donnelley was able to establish the effectiveness of potential remedies before any work began. So, the company knew exactly which solution to select: rewriting the collator's control programs (about $30,000) to yield a 200% increase in throughput. Donnelley then used the simulation to further optimize the collator, proving out the design for future expansion.

"Most software packages these days come with more than one disk," explains Michael Savoldi, engineering facilitator for Donnelley. "So when you buy software in a store, it's nice to have all the disks inside, and the right ones at that. Missing disks are more critical than duplicate disks."

Before 1990, assembling these software packages for retail sales was strictly a manual, job-shop operation. Employees hand-collated the disks, put them in tamper-proof packages, and placed these packages with their published documentation into a box. Finally, they would shrink-wrap the box, if necessary, and ship this "assembly" out through sales.

Donnelley decided to automate this process with a customized disk collator. In operation, prelabeled disks are loaded into various hoppers on the collator. A vision system monitors each disk as it leaves the hoppers. Correct disks go on to be collated; incorrect disks are rejected and reloaded into the appropriate hopper. Disks are collated into groups of up to 16 disks in 3¼- and 5½-in. sizes, then sealed and matched with the printed documentation. The collator can assemble several thousand of these software packages every hour.

The collator consists of three conveyors: disk end feed conveyor, disk container delivery conveyor, and finished subassembly conveyor, which takes disk packages to the sealer. It also has associated motor starters, diverters, picker arms, a vision system, and other control devices. An Allen-Bradley Co. (Milwaukee, WI) SLC-500 programmable controller, with approximately 250 discrete input/output (I/O) points, directs the automated devices on the collator. This one-of-a-kind collator was installed at Donnelley in early 1991, without as-built documentation and operating below spec.

"The machine would constantly have mismatches and rejects, the strobe light didn't work well, and a lot of motor fault problems occurred. In short, it was mechanically and functionally unacceptable," explains Savoldi. "I needed to do something right away to prove that this machine would work, and I didn't have the engineering, PLC programming, or simulation support in-house to help."

Savoldi contracted HEI Corp. (Carol Stream, IL) to evaluate the collator's operation using Automation Master, Donnelley's simulation/emu-

A monitor (foreground) showing a simulation of R.R. Donnelley's disk collator (background), which displays the operating status of the machine.

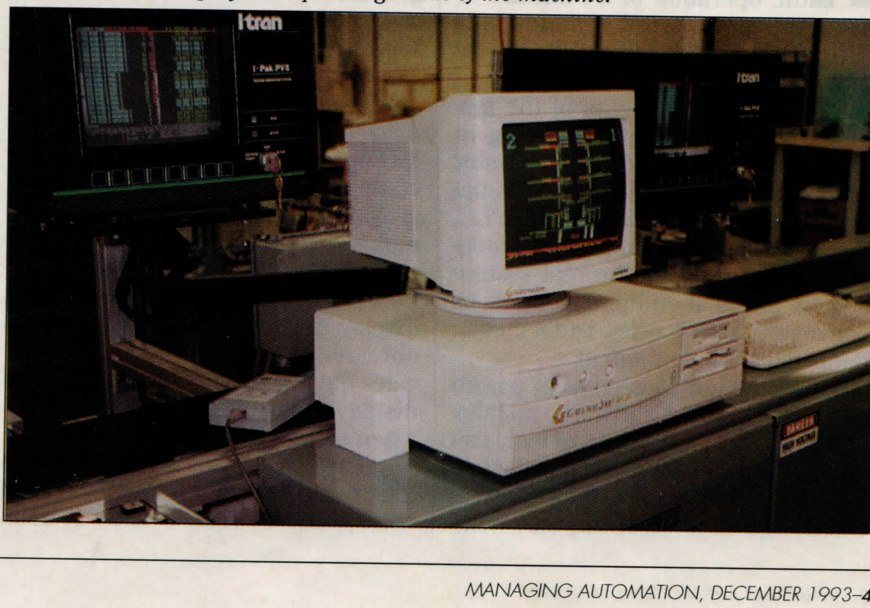

Photo courtesy: R.R. Donnelley & Sons

Janet Endrijonas is the editor of the Software section. She can be reached at (714) 240-0214 or fax (714) 240-0346, P.O. Box 5704, San Clemente, CA 92674.

The Year of the Operating System

93
YEAR IN REVIEW

Until late in the summer, Windows NT remained the world's greatest vaporware story. Jokes about "NT" really meaning "not there" were rampant, and even those who smiled smugly because they had a beta copy weren't giving anything away. Then one morning NT became reality—available in the marketplace—and both OS/2 and UNIX suddenly sat up and took notice.

One of the reasons that OS/2 has had continual difficulty carving a market niche has been the dearth of available applications. While there will certainly be new applications shipping for OS/2, their numbers will be nowhere near the volume of new applications coming out for NT. As always in this business, what I am saying this year may come back to haunt me. By this time next year, OS/2 third-party developers could conceivably be churning out software at an incredible rate—but if historical perspective is of any value, this isn't likely.

On the other hand, UNIX, despite its ongoing inability to settle on a single version, is not about to steal silently off into the sunset. "Users have been moving to UNIX for years because it is perceived as an open operating system," says Peter West, vice president of marketing for the Manufacturing Business Unit of the ASK Group (Mountain View, CA). "In the past, manufacturers have looked at the systems in their companies—mainframe plus minis, plus UNIX workstations, multiple databases and operating systems—and they have seen UNIX as the unifying platform because it is vendor independent."

Windows NT embraces the theme of unifying in a way UNIX cannot because an enormous number of the installed base of PCs are operating on DOS or MS/Windows. West foresees what Microsoft has already announced—that future versions of MS/Windows will evolve that product closer to NT as years go by, until they eventually are one and the same. Just as users have demanded UNIX, so will they begin to demand NT. NT will become the LAN and client operating system while OS/2 and UNIX will remain server software.

People who are moving away from mainframes will still go to UNIX, predicts West, who still questions how fast NT will make inroads into UNIX territory on the server. And, porting will not be the only answer. As easy as porting may be, moving from UNIX to NT requires a Windows interface that UNIX doesn't have. In addition, there will be plenty of companies that will find it difficult to port their applications to NT, especially depending upon what tools were used to build those applications. In this case, it will probably result in a "if it ain't broke, don't fix it" attitude.

Of course operating systems weren't the only type of software that made news in 1993. Manufacturing execution systems (MES) came into their own this year as one of the types of software necessary to ensure manufacturing success. MES software resides on the plant floor between the planning upstairs and the data collected in the process itself, turning the latter into more usable information. MES software bridged a gap that had always existed and integrated the enterprise even more closely.

CALS (Computer-Aided Acquisition and Logistics Support) software, while still heavily tied to its birth at the Department of Defense, finally began to make some headway into the civilian manufacturing realm in the U.S.—something that had already happened in Europe. In the summer, Computer Sciences Corp. (El Segundo, CA) made the first noteworthy sale of its E-CALS (Enterprise-CALS) product to civilian industry. Other manufacturers, for example Caterpillar Inc. (Peoria, IL), had already adopted the E-CALS concept and everyone knows that if it plays in Peoria...

Artificial intelligence and expert systems software became more application-specific. They began shedding their somewhat empty shells and making installation easier, thus providing a more recognizable framework into which a finished system for a specific application could fit. This may not seem to make sense for a classification of software so closely tied to the way in which any given company uses information, but the developers have managed to add a new flexibility that makes the use of this software a reality for more companies.

There was much more that developed in manufacturing software in 1993—there just isn't room to talk about it all. And 1994 promises to be another big year. Not only will software developers look for ways to help industry compete in the global economy, but the operating-systems vendors each tout their own platform, while encouraging developers to write these programs as fast as possible in order to appeal to the end user. After all, the operating system that can provide the best solution with the most applications will ultimately win the market race. While the ultimate victor will be determined by market demand, the operating system with the most to offer will stand the best chance to foster that demand and tilt the market in its own favor.

By Janet Endrijonas

lation system that Donnelley had used successfully on previous projects.

Automation Master is a DOS-based, high-resolution, real-time simulation system that mimics the operation of a machine or an entire automated system. It is "programmed" by building a model that describes the characteristics of the physical components of the machine or system. Then, running in emulation mode, Automation Master can be connected directly to a machine's or system's controller—PLC, computer, or whatever. The emulation model replaces the real system and its physical equipment, such as conveyors, fork trucks, robots, even data collection equipment.

Automation Master tests design concepts, control software, installation, and operations. Control logic can be fully tested before field installation, to see if the control software meets the design criteria. It can also compare an installed "as-built" with design specifications, and experiment with and evaluate different modifications to control logic and physical setup.

As a simulator, Automation Master displays detailed, dynamic color graphics with up to 1,500 variables and 1,400 entities. It can output data in WK1 (a standard spreadsheet format), run in real or accelerated time, and offers overview and zoom capability. As an emulator, it can test up to 32,767 I/O points per model, simultaneously communicate with up to four PLCs, provide analog representation, and simultaneously emulate automatic operation and accept inputs from a keyboard. As a monitor, the system's display pinpoints bottlenecks, prints out audit trails and failure logs, indicates out-of-tolerance equipment, and displays plain English descriptions when identifying failures.

The collator model was based on the original design documentation and "best guesses" of the collator as built. The data entered to make the model included both collator physical makeup and its operating variables, such as the cycle times and conveyor, pickup arms, vision system, and ejector speeds. Automation Master was then connected to the PLC controlling the collator. The differences in operation between the simulation and the collator were logged to a printer. As required, the simulation was changed to eliminate the differences between the as-built model and the actual collator, resulting in a true, field-verified, real-time software model.

Donnelley's first need was for the collator to operate at design speed. Some outside consultants suggested the problem was in the vision system, which perhaps was not processing images of the disk identification labels fast enough. Their $200,000 fix was expected to double collator speed.

Before making that investment, the operating parameters of the proposed vision system were entered into

Automation Master. "We found that the vision acquisition was the fastest thing happening: snap the picture, compare it, give a yes/no, pass/fail determination," explains Savoldi. "It was the mechanical part of the collator that couldn't function fast enough to keep up with the vision system." In fact, simulation showed that changes to the vision system would yield only a 20% increase in machine capacity.

"We simulated other remedies, played what-if games, and put the results into a spreadsheet for analysis," says Max Hitchens, president of HEI. "We figured that by changing the sequence of operations and overlapping some cycles—rather than have operations run sequentially, which was a much cheaper solution—we could double the collator's throughput."

Donnelley now uses Automation Master to debug and refine the collator for particular jobs, as well as to act as an on-line maintenance monitor.

"Automation Master provided us with an electronic version of our disk collator," says Savoldi. "We can watch it do whatever it would do on screen. Then we can electronically correct the collator and investigate modifications without investing any money into actually rebuilding the collator.

"If it hadn't been for this simulation system," he sums up, "we could have invested in an unproven scheme that could have made the collator worse." MA

HEI Corp.
Automation Master .... **RC# 137**

See Information Express.